7.1 DISTRIBUTED DATABASES

(GGSIPU, 2011; MDU, Dec, 2009, May 2009-10, KU)

Distributed databases can be termed as collection of multiple databases that are stored on several computers across various location connected to one another through a computer network.

A user sees a distributed database as a single database which is located on a single computer. He does not have any idea that the particular data which he is accessing may be located at some other site. A distributed database management system is a set of programme that uses client-server architecture to process information requests.

7.1.1 Types of Distributed Databases

There are two types of distributed databases:

(i) **Homogeneous distributed databases:** Databases stored at various geographical regions runs identical database softwares.

(ii) **Heterogeneous distributed databases:** Databases stored at various geographical regions have different database softwares. For example, one site may be running oracle database while other may have DB2 database.

7.1.2 Design of Distributed Database

Different techniques used for designing distributed databases are:

1. **Data Fragmentation:** In this technique, decision is made regarding what portion of database is to be stored at which location. A relation is broken into different fragment and is physically stored across various sites. Various ways of fragmenting a relation are:
   (a) ***Horizontal Fragmentation:*** A relation R is partitioned into many relations where each new relation consists of some tuples of relation R. These new relations are distributed across various sites.
   Example: Consider the following student relation

Student

Roll No	Name	Branch	Marks
1	Ashu	CSE	95
2	Binoy	CSE	84
3	Himanshu	IT	79
4	Naina	CSE	70
5	Rashmi	IT	65

Fig. 7.1: Student Relation

This relation can be partitioned according to branch field of a student i.e. as follows:

$$\text{Student\_Frag1} = \sigma_{\text{Branch = 'CSE'}}(\text{Student})$$
$$\text{Student\_Frag2} = \sigma_{\text{Branch = 'IT'}}(\text{Student})$$

Student_Frag1

Roll No	Name	Branch	Marks
1	Ashu	CSE	95
2	Binoy	CSE	84
4	Naina	CSE	70

Student_Frag2

Roll No	Name	Branch	Marks
3	Himanshu	IT	79
5	Rashmi	IT	65

Fig. 7.2: Horizontal Fragmentation

(b) ***Vertical Fragmentation:*** A relation R is partitioned into many relations r_i where each new relation consist of only certain attributes of a relation R. An additional attribute Tuple_Id is added which specifies logical or physical address of a tuple.
Example: The student relation is partitioned into two new relations, one relation contains RollNo., Marks while the other contains Name and Branch fields of a student.

$$\text{Student\_Vfrag1} = \pi_{\text{rollno, marks, Tuple\_Id}}(\text{Student})$$
$$\text{Student\_Vfrag2} = \pi_{\text{name, branch, Tuple\_ID}}(\text{Student})$$

Student_Vfrag1

Roll No	Marks	Tuple_Id
1	95	1
2	84	2
3	79	3
4	78	4
5	65	5

Student_Vfrag2

Name	Branch	Tuple_Id
Ashu	CSE	1
Binoy	CSE	2
Himanshu	IT	3
Naina	CSE	4
Rashmi	IT	5

Fig. 7.3: Vertical Fragmentation

(c) ***Mixed Fragmentation:*** In this type of fragmentation a relation is first partitioned horizontally and then the new relation obtained is further

partitioned vertically or a relation is first partitioned vertically and then the new relation obtained is further partitioned horizontally.

Example:

$$Stud = \pi_{RollNo,\ Name}\ (\sigma_{Branch\ =\ CSE})(Student)$$

Stud.

RollNo	Name
1	Ashu
2	Binoy
4	Naina

Fig. 7.4: Mixed Fragmentation

2. **Data Replication:** It refers to maintaining of more than one copy of a data at several different site i.e. many identical replicas of a relation is stored at more than one site.
   Two types of data replications are:
   (a) ***Fully Replicated Database:*** A copy of entire database is replicated at more than one site.
   (b) ***Partially Replicated Database:*** Some portion of a database is replicated at other site.
3. ***Data Allocation:*** Data allocation is a strategy by which one decides how to place data at different site. In centralised strategy data and DBMS is stored at a single site and users at different site can access this data through a network. Another strategy is to partition the data and store them at different site or keep different copies of same data at several sites.

7.1.4 Advantages of Distributed Databases

1. **Efficient management of data with different level of transparency:** Transparency means hiding certain details from users about where the data is actually stored.
   Different types of transparencies are:
   (a) ***Distribution transparency:*** The user perceive a picture of distributed database as a single database stored on a single computer. He is not aware of the fact that the data may be located at different locations. This type of transparency can be further divided into—
      (i) ***Location transparency***: The command use to access a particular data is independent of the actual physical location of data. Users have to just give a command on the computer on which he is working. Internally routing of the command to actual physical location of data and bringing back the result is the work of distributed database system.
      (ii) ***Naming transparency***: No extra information is to be provided by a user to access a data which is located on some other site. It is the responsibility of distributed database to see whether no two sites have a database object with same name.
   (b) ***Replication transparency:*** A data object may be replicated at several different site for better availability. The user of a system need not be aware about this fact.
   (c) ***Fragmentation transparency:*** User need not be aware of the fact that a relation may have been fragmented horizontally or vertically at different sites.
2. **Improved reliability and availability:** Since a query can be processed by a computer on any site, so even if one site fails then other continues to work.
3. **Easier expansion:** More and more sites can easily be added.
4. **High performance:** Distribution of data at various site is done in such a manner that the data is placed closer to where it is used frequently. This is referred to as data localisation.
5. **Parallel evaluation:** A query can be sub-divided and executed at various site.

7.1.5 Disadvantages of Distributed Databases

1. Difficult to maintain integrity. Since many replica of a data can be placed at various site, it is difficult to check whether all the replicas are consistent whenever an update is done.
2. High software development cost.
3. Technical problems of connecting dissimilar machine.
4. Complexity in designing and implementation of a system.
5. Increased processing overhead.
6. Security problem as there are many replica of data at different site.